

IN THE CLAIMS:

1 1. (Previously Presented) A method for establishing identity in a file system,
2 comprising:
3 receiving, from a client, a first Network File System (NFS) operation concerning
4 an indicated file, the first NFS operation received by a proxy;
5 forwarding the first NFS operation from the proxy to be received by a file server;
6 returning a NFS file handle associated with the first NFS operation from the file
7 server to the proxy in response to the file server receiving the first NFS operation from
8 the proxy;
9 inserting, by the proxy, metadata into the NFS file handle in response to receiving
10 the NFS file handle from the file server, wherein the metadata is an encryption key;
11 sending, by the proxy in response to receiving the NFS file handle from the file
12 server, the NFS file handle with the metadata inserted in the NFS file handle to the client
13 as a reply to the first NFS operation;
14 using, by the client, the metadata and the NFS file handle in a second NFS
15 operation to identify the client and the indicated file; and
16 receiving, from the client, the second NFS operation by the proxy, the second
17 NFS operation comprising the metadata sent with the second NFS operation;
18 identifying, in response to the metadata, the client as having a permission to
19 submit the second NFS operation;
20 sending the second NFS operation to the file server and not sending the metadata
21 to the file server; and
22 receiving, by the proxy, a further NFS reply from the file server, and sending, by
23 the proxy, the further NFS reply to the client.

1 2. (Previously Presented) The method of Claim 1, whereby using the metadata in the
2 NFS file handle eliminates a need for the proxy to generate additional requests to the file
3 server to establish file identity, and for completing client requests.

- 1 3. (Previously Presented) The method of Claim 1, further comprising:
2 encoding metadata in a form of a session key into the file handle, the session key
3 expiring after a predetermined amount of time.
- 1 4. (Previously Presented) The method of Claim 1, further comprising:
2 using an NFS file system as the file system.
- 1 5. (Previously Presented) The method of Claim 1, further comprising:
2 using a stateless protocol by the file system.
- 1 6-29. (Cancelled).
- 1 30. (Previously Presented) The method of claim 1, wherein the NFS file handle is of a
2 variable size.
- 1 31. (Previously Presented) A method for establishing identity in a file system,
2 comprising:
3 receiving a first file request concerning an indicated file from a client, the first file
4 request received by a proxy;
5 forwarding the first file request from the proxy to a file server;
6 returning a reply associated with the first file request from the file server to the
7 proxy, wherein the reply includes a file handle associated with the indicated file;
8 inserting, by the proxy, metadata into the file handle;
9 sending, by the proxy, the file handle with the metadata inserted in the file handle
10 to the client, the metadata to be used in further requests to identify the client as having a
11 permission to access the indicated file;
12 receiving, from the client, a second file request by the proxy, the second file
13 request including the metadata in a second file handle sent with the second file request;
14 identifying, in response to the metadata, that the client has the permission to
15 submit the second file request;

16 sending the second file request to the file server and not sending the metadata
17 with the second file handle to the file server; and
18 receiving by the proxy a second reply from the file server, and sending by the
19 proxy the second reply to the client.

1 32. (Currently Amended) An apparatus to establish identity in a file system,
2 comprising:

3 a proxy configured to receive a first Network File System (NFS) operation
4 concerning an indicated file sent by a client to the file system, the proxy further
5 configured to forward the first NFS operation to be received by a file server;

6 the file server configured to return a NFS file handle associated with the first NFS
7 operation to the proxy in response to the file server receiving the first NFS operation
8 from the proxy;

9 the proxy further configured to insert metadata into the NFS file handle in
10 response to receiving the NFS file handle from the file server, wherein the metadata is an
11 encryption key; ~~and~~

12 the proxy further configured to send the NFS file handle with the metadata
13 inserted in the NFS file handle to the client as a reply to the first NFS operation, the
14 metadata and the NFS file handle to be used in a second NFS operation to identify the
15 client and the indicated file;

16 the proxy further configured to receive, by the client, a second NFS operation, the
17 second NFS operation comprising the metadata in the second NFS file handle sent with
18 the second NFS operation;

19 the proxy to identify, in response to the metadata, the client as having a
20 permission to submit the second NFS operation;

21 the proxy to send the second NFS operation to the file server and not to send the
22 metadata with the second NFS file handle to the file server; and

23 the proxy to receive a second NFS reply from the file server, and the proxy to
24 send the second NFS reply to the client.

1 33. (Currently Amended) The apparatus of Claim 32, whereby using the metadata in the
2 NFS file handle ~~eliminated the~~eliminates a need for the proxy to generate additional
3 requests to the file server to complete client requests.

1 34. (Previously Presented) The apparatus of Claim 32, further comprising:
2 the proxy to use the metadata in the NFS file handle received from the client to
3 eliminate a need for additional communication with the file server to establish file
4 identity.

1 35. (Previously Presented) The apparatus of Claim 32, further comprising:
2 the proxy to encode the metadata in a form of a session key into the NFS file
3 handle, the session key expiring after a predetermined amount of time.

1 36. (Previously Presented) The apparatus of Claim 32, further comprising:
2 an NFS file system used as the file system.

1 37. (Previously Presented) The apparatus of Claim 32, further comprising:
2 a stateless protocol used by the file system.

1 38. (Currently Amended) A non-volatile memory executed on a computer, comprising:
2 the non-volatile memory containing procedures for execution on the computer for
3 a method of establishing identity in a file system, the method having the steps of,
4 receiving, from a client, an operation concerning an indicated file, the ~~first~~
5 operation received by a proxy;
6 forwarding the ~~first~~ operation from the proxy to be received by a file server;
7 returning a file handle associated with the first operation from the file server to
8 the proxy in response to the file server receiving the ~~first~~ operation from the proxy;
9 inserting, by the proxy, metadata into the file handle in response to receiving the
10 NFS file handle from the file server, wherein the metadata is an encryption key; and

11 sending, by the proxy in response to receiving the file handle from the file server,
12 the file handle with the metadata inserted in the file handle to the client as a reply to the
13 ~~first~~ operation;
14 receiving, from the client, a second file request by the proxy, the second file
15 request comprising the metadata in a second file handle sent with the second file request;
16 identifying, in response to the metadata, that the client has permission to submit
17 the second file request;
18 sending the second file request to the file server and not sending the metadata
19 with the second file handle to the file server; and
20 receiving, by the proxy, a second reply from the file server, and sending by the
21 proxy the second reply to the client.

1 39. (Currently Amended) A method for establishing identity in a file system,
2 comprising:
3 receiving a first file request concerning an indicated file from a client, the first file
4 request received by a proxy;
5 forwarding the first file request from the proxy to a file server;
6 granting a permission for the request to be acted upon by the file system in
7 response to a predetermined protocol;
8 returning a reply associated with the first file request from the file server to the
9 proxy, wherein the reply includes a file handle associated with the indicated file;
10 inserting, by the proxy, a session key into the file handle;
11 sending, by the proxy, the file handle with the session key inserted in the file
12 handle to the client, the session key to be used in further requests to identify the client
13 and the indicated file;
14 receiving, from the client, a second file request by the proxy, the second file
15 request comprising information from the session key in a second file handle sent with the
16 second file request;
17 identifying, in response to the session key, that the client has the permission to
18 submit the second file request;

19 sending the second file request to the file server and not sending the session key
20 with the second file handle to the file server; and
21 receiving, by the proxy, a second reply from the file server, and sending by the
22 proxy the second reply to the client.

1 40. (Previously Presented) The non-volatile memory of Claim 38, whereby using the
2 metadata in the file handle eliminates a need for the proxy to generate additional requests
3 to the file server to establish file identity.

1 41. (Previously Presented) The non-volatile memory of Claim 40, further comprising:
2 causing the session key to expire after a selected amount of time.

1 42. (Previously Presented) The non-volatile memory of Claim 40, further comprising:
2 causing the session key to expire after a selected amount of usage.

1 43. (Previously Presented) The non-volatile memory of Claim 38, further comprising:
2 using a NFS file server as the file server.

1 44. (Previously Presented) The non-volatile memory of Claim 38, further comprising:
2 using a two way communication exchange between the proxy and the file server.

1 45. (Currently Amended) An apparatus to establish identity in a file system, comprising:
2 a proxy to receive a file request sent by a client to the file system, the proxy to
3 forward the request to a file server;
4 the file server to return a reply associated with the file request to the proxy,
5 wherein the reply includes a file handle;
6 the proxy to insert a session key into the file handle;
7 the proxy to send the file handle with the session key inserted in the file handle to
8 the client, the session key to be used in further requests to identify the client and the
9 indicated file;

10 the proxy to receive, by the client, a second file request, the second file request to
11 | include information of the session key in a further file handle sent with the second
12 request;

13 | the proxy to identify, in response to the information of the session key, the client
14 as having a permission to submit the another file request;

15 the proxy to send the second request to the file server and not to send the session
16 key with the second file handle to the file server; and

17 the proxy to receive a further reply from the file server, and the proxy to send the
18 further reply to the client.

1 46. (Previously Presented) The apparatus as in claim 45, whereby using the session key
2 in the file handle eliminates a need for the proxy to generate additional requests to the file
3 server to establish file identity.

1 47. (Previously Presented) The apparatus of Claim 45, wherein the file handle is a
2 Network File System (NFS) file handle.

1 48. (Previously Presented) The apparatus of Claim 45, further comprising:
2 the proxy to encode the metadata in a form of a session key into the file handle,
3 the session key expiring after a predetermined amount of time.

1 49. (Previously Presented) The apparatus of Claim 45, further comprising:
2 an NFS file system used as the file system.

1 50. (Previously Presented) The apparatus of Claim 45, further comprising:
2 a stateless protocol used by the file system.

1 51. (Previously Presented) An apparatus to establish identity in a file system,
2 comprising:

3 a proxy configured to receive a first file request sent by a client to the file system,
4 the proxy further configured to forward the first file request to a file server;
5 the file server configured to return a reply associated with the first file request to
6 the proxy;
7 the proxy further configured to insert a session key into a file handle;
8 the proxy further configured to send the file handle with the session key inserted
9 in the file handle to the client, the session key configured to be used in a second file
10 request to identify the client and the indicated file;
11 the proxy further configured to receive, by the client, a second file request, the
12 second file request configured to include the session key in a second file handle sent with
13 the second file request;
14 the proxy further configured to identify, in response to the session key, the client
15 as having a permission to submit the second file request;
16 the proxy further configured to send the second file request to the file server and
17 not to send the session key with the second file handle to the file server; and
18 the proxy further configured to receive a second reply from the file server, and the
19 proxy further configured to send the second reply to the client.

1 52. (Currently Amended) A method for establishing identity in a file system,
2 comprising:

3 receiving a first file request concerning an indicated file from a client, the first file
4 request received by a proxy;
5 forwarding the first file request from the proxy to a file server;
6 determining that the client has a permission to have the request acted upon by the
7 file system in response to a predetermined protocol;
8 returning a reply associated with the first file request from the file server to the
9 proxy, wherein the reply includes a file handle associated with the indicated file;
10 inserting, by the proxy, a cryptographic information into the file handle;

11 sending, by the proxy, the file handle with the cryptographic information inserted
12 in the file handle to the client, the cryptographic information to be used in one or more
13 requests to identify the client and the indicated file;

14 receiving, by the client, a second file request by the proxy, the second file request
15 including the cryptographic information in a second file handle sent with the second file
16 request;

17 identifying, in response to the cryptographic information, that the client has the
18 permission to submit the second file request;

19 sending the second file request to the file server and not sending the cryptographic
20 information with the second file handle to the file server; and

21 receiving, by the proxy, a second reply from the file server, and sending by the
22 proxy the second reply to the client.

1 53. (Previously Presented) The method according to claim 52, whereby using the
2 cryptographic information in the file handle eliminates a need for the proxy to generate
3 additional requests to the file server to establish file identity.

1 54. (Previously Presented) The method according to claim 52, further comprising:
2 causing the cryptographic information to expire after a selected amount of time.

1 55. (Previously Presented) The method according to claim 52, further comprising:
2 causing the cryptographic information to expire after a selected amount of usage.

1 56. (Previously Presented) The method according to claim 52, further comprising:
2 using a NFS protocol as the predetermined protocol.

1 57. (Previously Presented) The method according to claim 52, further comprising:
2 using as the predetermined protocol a two way communication exchange between
3 the proxy and the file server.

1 58. (Previously Presented) An apparatus to establish identity in a file system,
2 comprising:

3 a proxy configured to receive a file request for an indicated file sent by a client to
4 the file system, the proxy further configured to forward the request to a file server;

5 the file server configured to return a reply associated with the file request to the
6 proxy, wherein the reply is configured to include a file handle;

7 the proxy further configured to insert a cryptographic information into the file
8 handle;

9 the proxy further configured to send the file handle with the cryptographic
10 information inserted in the file handle to the client, the cryptographic information
11 configured to be used in further requests to identify the client and the indicated file;

12 the proxy further configured to receive, by the client, a second request, the second
13 file request to include the cryptographic information in a second file handle sent with the
14 second request;

15 the proxy further configured to identify, in response to the cryptographic
16 information, the client as having a permission to submit the second file request;

17 the proxy further configured to send the second request to the file server and not
18 to send the cryptographic information with the second file handle to the file server; and

19 the proxy further configured to receive a further reply from the file server, and the
20 proxy to send the further reply to the client.

1 59. (Previously Presented) The apparatus as in claim 58, whereby using the
2 cryptographic information in the file handle eliminates a need for the proxy to generate
3 additional requests to the file server to establish file identity.

1 60. (Previously Presented) The apparatus of claim 58, wherein the file handle is a
2 Network File System (NFS) file handle.

1 61. (Previously Presented) The apparatus of claim 58, further comprising:
2 the proxy further configured to encode the metadata in a form of a cryptographic
3 information into the file handle, the cryptographic information configured to expire after
4 a predetermined amount of time.

1 62. (Previously Presented) The apparatus of claim 58, further comprising:
2 an NFS file system used as the file system.

1 63. (Previously Presented) The apparatus of claim 58, further comprising:
2 a stateless protocol used by the file system.

1 64. (Previously Presented) An apparatus to establish identity in a file system,
2 comprising:
3 a proxy configured to receive a first file request sent by a client to the file
4 system, the proxy to forward the first file request to a file server;
5 the file server configured to return a reply associated with the first file request
6 to the proxy;
7 the proxy further configured to insert a cryptographic information into a file
8 handle;
9 the proxy further configured to send the file handle with the cryptographic
10 information inserted in the file handle to the client, the cryptographic information
11 configured to be used in a second file request to identify the client and the indicated
12 file;
13 the proxy further configured to receive, by the client, a second file request, the
14 second file request configured to include the cryptographic information in a second
15 file handle sent with the second file request;
16 the proxy further configured to identify, in response to the cryptographic
17 information, the client as having a permission to submit the second file request;

18 the proxy further configured to send the second file request to the file server
19 and not to send the cryptographic information with the second file handle to the file
20 server; and
21 the proxy further configured to receive a second reply from the file server, and
22 the proxy to send the second reply to the client.

1 65. (Previously Presented) A method for establishing identity in a file system,
2 comprising:
3 receiving a file request concerning an indicated file from a client, the request
4 received by a proxy;
5 forwarding the request from the proxy to a file server;
6 returning a reply associated with the file request from the file server to the
7 proxy, wherein the reply includes a file handle associated with the indicated file;
8 inserting, by the proxy, metadata into the file handle;
9 sending, by the proxy, the file handle with the metadata inserted in the file
10 handle to the client, a size of the file handle set to a sum of a length of the server file
11 handle and a length of the proxy metadata, the metadata to be used in further requests
12 to identify the client and the indicated file; and
13 receiving, from the client, a second file request by the proxy, the second file
14 request comprising the metadata in a second file handle sent with the second file
15 request;
16 identifying, in response to the metadata, that the client has permission to
17 submit the second file request;
18 sending the second file request to the file server and not sending the metadata
19 with the second file handle to the file server; and
20 receiving by the proxy a second reply from the file server, and sending by the
21 proxy the second reply to the client.

1 66. (Previously Presented) A method, comprising:
2 receiving, by a proxy, a file request for a file sent from a client;

3 forwarding the file request from the proxy to a file server;
4 returning a reply associated with the file request from the file server to the
5 proxy, wherein the reply includes a file handle;
6 inserting, by the proxy, metadata into the file handle;
7 sending, by the proxy, the file handle with the metadata inserted in the file
8 handle to the client;
9 receiving, from the client, a second file request by the proxy, the second file
10 request comprising the metadata in a second file handle sent with the second file
11 request;
12 identifying, in response to the metadata, that the client has permission to
13 submit the second file request;
14 sending the second file request to the file server and not sending the metadata
15 with the second file handle to the file server; and
16 receiving by the proxy a second reply from the file server, and sending by the
17 proxy the second reply to the client.

1 67. (Currently Amended) A computer apparatus, comprising:
2 a proxy configured to receive a client file request for a file and forward the file
3 request from the proxy to a file server;
4 the server configured to return a reply associated with the file request, wherein the
5 reply includes a file handle;
6 the proxy further configured to intercept the file handle sent from the server and
7 insert metadata into the file handle to create a modified file handle;
8 the proxy further configured to send the modified file handle with the metadata
9 inserted in the file handle to the client;
10 the proxy further configured to receive the modified file handle from the client for
11 a second file request for the file, wherein the proxy is further configured to use the
12 modified file handle to eliminate a need for the proxy to generate one or more additional
13 requests to the server that would be required to access the file if the modified file handle
14 did not include the inserted metadata;

15 | the proxy further configured to receive, by the client, a second file request, the
16 second file request configured to include the metadata in a second file handle sent with
17 the second file request;
18 the proxy further configured to identify, in response to the metadata, the client as
19 | having a permission to submit the second file request;
20 the proxy further configured to send the second file request to the file server and
21 not to send the metadata with the second file handle to the file server; and
22 the proxy further configured to receive a second reply from the file server, and the
23 proxy to send the second reply to the client.